Journal of Computational Acoustics,

Adaptive Finite Element Techniques for the Acoustic Wave Equation*

Wolfgang Bangerth and Rolf Rannacher
Institute of Applied Mathematics, University of Heidelberg, Germany
{wolfgang.bangerth,rannacher}@iwr.uni-heidelberg.de

Received (to be inserted
Revised by Publisher)

We present an adaptive finite element method for solving the acoustic wave equation. Using a global duality argument and Galerkin orthogonality, we derive an identity for the error with respect to an arbitrary functional output of the solution. The error identity is evaluated by solving the dual problem numerically. The resulting local cell-wise error indicators are used in the grid adaptation process. In this way, the space-time mesh can be tailored for the efficient computation of the quantity of interest. We give an overview of the implementation of the proposed method and illustrate its performance by several numerical examples.

1. Introduction

We consider the acoustic wave equation

$$\begin{aligned} \rho(\mathbf{x})u_{tt}(\mathbf{x},t) - \nabla\cdot(a(\mathbf{x})\nabla u(\mathbf{x},t)) &= 0 && (\mathbf{x},t)\in Q_T=\Omega\times I,\\ u(\mathbf{x},0)=u_0(\mathbf{x}),\ \ u_t(\mathbf{x},0)&=v_0(\mathbf{x}) && \mathbf{x}\in\Omega,\\ \mathbf{n}\cdot a(\mathbf{x})\nabla u(\mathbf{x},t) &= 0 && (\mathbf{x},t)\in\Gamma\times I,\Gamma=\partial\Omega. \end{aligned} \tag{1.1}$$

on a space-time region $Q_T := \Omega \times I$, where $\Omega \subset \mathbb{R}^d$, $d \geq 1$, and $I = (0,T)$; the density ρ and elastic coefficient a may vary in space. This equation frequently occurs in the simulation of acoustic waves in gaseous or fluid media, seismics, electrodynamics and many other applications. The extension to Dirichlet and non-homogeneous boundary conditions is straightforward but will be omitted for brevity; details for these cases can be found in reference [1].

We will solve eq. (1.1) by a "velocity-displacement" formulation which is obtained by introducing a new velocity variable $v = u_t$. Then, the natural solution space is $W := H^1(I,H^1(\Omega)) \times H^1(I,L_2(\Omega))$. The corresponding weak formulation is obtained by multiplying by test functions from the "test space" $T := \left\{\boldsymbol{\tau} : I \to L^2(\Omega)\times H^1(\Omega), \boldsymbol{\tau} \textit{ piecewise smooth in time in } [0,T]\right\}$, integrating by parts in space and imposing initial values

*Presented at ICTCA'99 the 4th International Conference on Theoretical and Computational Acoustics, May 1999, Trieste, Italy

$\mathbf{w}_0 = (u_0, v_0)$ in a weak sense: Find $\mathbf{w} = (u, v) \in W$, such that

$$a(\mathbf{w}, \boldsymbol{\tau}) = (\rho \mathbf{w}_0, \boldsymbol{\tau}(0))_\Omega \qquad \forall \boldsymbol{\tau} = (\varphi, \psi) \in T, \tag{1.2}$$

with the bilinear form

$$a(\mathbf{w}, \boldsymbol{\tau}) = (\rho \mathbf{w}_t, \boldsymbol{\tau})_{Q_T} + \left(\begin{pmatrix} 0 & -\rho \\ a\nabla & 0 \end{pmatrix} \mathbf{w}, \begin{pmatrix} 1 & 0 \\ 0 & \nabla \end{pmatrix} \boldsymbol{\tau} \right)_{Q_T} + (\rho \mathbf{w}(0), \boldsymbol{\tau}(0))_\Omega . \tag{1.3}$$

By definition of W and T, the evaluation of $\mathbf{w}$ and $\boldsymbol{\tau}$ at a discrete time is well-defined.

Here and throughout the paper, we will use the following notation:

- $(\varphi, \psi) = \int \varphi\psi \; dx\, dt$ L_2 scalar product; we also use the obvious extension to vector valued functions φ and ψ;
- $\|\varphi\| = (\varphi, \varphi)^{1/2}$ L_2 norm;
- Time slabs: We subdivide the time interval $I = (0, T)$ into time slabs $I_n = (t_{n-1}, t_n)$ with length $k_n = t_n - t_{n-1}$. On each I_n, we use triangulations $\mathbb{T}^n$ satisfying the usual regularity condition (see, e.g., reference [2]); to ease mesh refinement and coarsening, we allow one "hanging node" per edge or face.

The outline of this paper is as follows: In Section 2, we describe the discretization of problem (1.2). Then, in Section 3, we outline the derivation of error estimators and in Section 4 the principles of an algorithm for mesh adaptation based on these estimates. In Section 5, we present results of some test computations and finally, in Section 6, state some open problems.

2. Discretization

To discretize problem (1.2), we use the so–called Rothe approach of first discretizing in time and then in space on each discrete time level. This has the advantage of having the freedom to choose the computational mesh differently at each time level. The time semi-discretization is performed using the Crank-Nicolson scheme:

$$\begin{aligned} \left(\rho u^n - \rho u^{n-1}, \varphi\right)_\Omega - \tfrac{1}{2} k_n \left(\rho v^n + \rho v^{n-1}, \varphi\right)_\Omega = 0, \\ \left(\rho v^n - \rho v^{n-1}, \psi\right)_\Omega + \tfrac{1}{2} k_n \left(a\nabla(u^n + u^{n-1}), \nabla\psi\right)_\Omega = 0, \end{aligned} \tag{2.4}$$

with $u^0 = u_0$ and $v^0 = v_0$. We chose this time stepping scheme because it is of second order and it is energy conserving, i.e.

$$\|\sqrt{\rho} v^n\|^2 + \left\|\sqrt{a}\nabla u^n\right\|^2 = \left\|\sqrt{\rho} v^{n-1}\right\|^2 + \left\|\sqrt{a}\nabla u^{n-1}\right\|^2.$$

This conservation property carries over to the the space-discretized equations provided that the meshes do not change between time levels.

Since we use a primal formulation of the problem, the space discretization of (2.4) can be based on any one of the usual conforming Lagrangian finite element spaces $Q^r(\mathbb{T}^n)$ of

degree r, which are defined on the computational grid $\mathbb{T}^n$ at time level t_n; these meshes may vary between the time levels in order to allow for grid refinement moving with the wave field. We use decompositions of Ω into quadrilaterals ($d = 2$) or hexahedra ($d = 3$). This discretization leads to a system of two coupled matrix equations, which can be rewritten in the form

$$\begin{aligned} \left(M + \tfrac{1}{4}k_n A\right) U^n &= MU^{n-1} + k_n MV^{n-1} - \tfrac{1}{4}k_n^2 AU^{n-1}, \\ MV^n &= MV^{n-1} - \tfrac{1}{2}k_n A\left(U^n + U^{n-1}\right). \end{aligned} \tag{2.5}$$

Here, $M = (M_{ij})_{ij} = \big((\rho\varphi_i, \varphi_j)\big)_{ij}$ is the "mass matrix" and $A = (A_{ij})_{ij} = \big((a\nabla\varphi_i, \nabla\varphi_j)\big)_{ij}$ the "stiffness matrix" corresponding to the nodal basis functions $\varphi_i, \varphi_j \in Q^r(\mathbb{T}^n)$.

This system splits into two equations, a discrete Helmholtz equation and a discrete L^2-projection, which are solved by a preconditioned conjugated gradient method; we use symmetric Gauss-Seidel or multigrid as preconditioner.

For the approach to error estimation which we will present in the next section, it is important to note that the above discretization in time and space can be interpreted as a simultaneous Galerkin discretization in space-time. We can formulate it is follows: Find $\mathbf{w}_h = (u_h, v_h) \in \mathcal{W}_h$, such that

$$a(\mathbf{w}_h, \boldsymbol{\tau}_h) = (\rho\mathbf{w}_0, \boldsymbol{\tau}_h(0))_\Omega \qquad \forall \boldsymbol{\tau}_h = (\varphi_h, \psi_h) \in \mathcal{T}_h. \tag{2.6}$$

Functions in $\mathcal{W}_h$ and $\mathcal{T}_h$ are functions of the time and space variables. We can recover the time-discretization by the Crank-Nicolson scheme and the space discretization by finite elements by a suitable choice of these function spaces; in the present case, the spatial components of these spaces are conforming finite element space $Q^r(\mathbb{T}^n)$ at each discrete time level, while the time component of $\mathcal{W}_h$ is piecewise linear and continuous in time and that of $\mathcal{T}_h$ piecewise constant and discontinuous in time. For more details on the construction of these spaces, we refer to references [1,3].

3. Error estimation

Traditional *a priori* error analysis usually yields error bounds like

$$\begin{aligned} \|u(\cdot,T) - u_h(\cdot,T)\|_\Omega \le C_1(T)k^2 \left\{ \sup_{0<t<T} \left\|\partial_t^3 u(\cdot,t)\right\|_\Omega + \sup_{0<t<T} \left\|\partial_t^2 \nabla u(\cdot,t)\right\|_\Omega \right\} + \\ + C_2(T)h^{r+1} \left\{ \sup_{0<t<T} \left\|\partial_t^2 \nabla^r u(\cdot,t)\right\|_\Omega + \sup_{0<t<T} \left\|\nabla^{r+1} u(\cdot,t)\right\|_\Omega \right\}, \end{aligned} \tag{3.7}$$

with h, k being the maximum space and time mesh size, respectively.[4] Similar estimates can be obtained for finite difference and other methods as well. However, (3.7) is not what one is usually interested in, since

- The constants C_i are unknown and estimates for them are usually too pessimistic.
- The exact solution u, which is unknown as well, enters the estimate.

- The required derivatives of the exact solution might not exist everywhere.
- The norm $\|u(\cdot,T) - u_h(\cdot,T)\|_\Omega$ might not be the quantity we want to control.

Estimate (3.7) only tells us that, if the exact solution is sufficiently smooth, then we can expect asymptotic convergence of (at least) second order. However, in practical applications, the required smoothness is often lacking and for complex equations like the wave equation with varying coefficients the actual quantitative behavior of the error may not show reduction by the forecasted rate for reasonable mesh sizes.

An error estimate usable in practice should therefore satisfy the following requirements:

- It should contain only computable constants.
- It should be an "a posteriori" estimate, i.e., it should make reference only to the computed solution u_h and to known data.
- It should not assume more regularity than can be realistically expected from the solutions.
- The error should be estimated with respect to any observable quantity of interest.

It will become apparent that one of these conditions needs to be sacrificed since otherwise we would have an exact error representation by which we could recover the exact continuous solution, which obviously is not possible. In practice, it will be shown that we make reference to another unknown function, being the continuous solution of a dual problem.

Another requirement for a practically useful error estimate is that it implies a method for systematically reducing the error by local mesh adaptation. One possibility would be that the error estimate yields cell-wise quantities which correlate with the contribution of this space-time cell to the error quantity considered. In that case, we could reduce the error by refining those cells with the largest error contributions. A global error representation without such localization is of limited value in practice since it does not offer an indicator for mesh refinement.

In the remainder of this section, we will first heuristically and then formally show how such an *a posteriori* error estimate can be constructed and how we can extract cell-wise refinement indicators. The derivation of these residual-based error estimates uses a method proposed by Becker and Rannacher[5] which is based on the general concept of *a posteriori* error estimation introduced by Eriksson and Johnson.[6]

3.1. *Heuristic derivation*

Assume that in an application, we are interested in the value of a linear output functional $J(u)$ of the solution, examples are listed in Table 1. Nonlinear functionals are allowed as well, but require some additional considerations the details of which can be found in reference [1]. We will call the functional for a quantity of interest the "target" or "error functional".

Table 1. Examples of target functionals (left) and their respective integral kernels (right).

Target functional	Integral kernel
$J(u) = u(\mathbf{x}_0, T),$	$j(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_0)\delta(t-T);$
$J(u) = \int_C u(\mathbf{x},T)\ ds,$	$j(\mathbf{x},t) = \delta_C(\mathbf{x})\delta(t-T);$
$J(u) = \int_{\Omega_0 \subset \Omega} u(\mathbf{x},T)\ dx,$	$j(\mathbf{x},t) = \chi_{\Omega_0}(\mathbf{x})\delta(t-T);$
$J(u) = \int_0^T u(\mathbf{x}_0,t)\ dt,$	$j(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_0).$

With each of these functionals an integral kernel $j(\mathbf{x},t)$ is associated, such that

$$J(u) = \int_{Q_T} j(\mathbf{x},t)u(\mathbf{x},t)\ dx\,dt.$$

As is obvious from the examples, j is often only a distribution rather than a regular function. In the case of nonlinear functionals, j will depend on the solutions u and u_h as well.

The goal of a simulation is to approximate $J(u)$ by $J(u_h)$ to best accuracy. We would like to obtain a bound on the error with respect to this target functional, $\mathcal{E} := J(u) - J(u_h)$, without the need to know the exact solution u. In order to achieve this, notice that due to the assumed linearity of the functional, this error can be represented as follows:

$$\mathcal{E} = J(u) - J(u_h) = J(u-u_h) = \int_{Q_T} j(\mathbf{x},t)\,(u(\mathbf{x},t) - u_h(\mathbf{x},t))\,dx\,dt.$$

Let now $L := \rho\partial_t^2 - \nabla\cdot(a\nabla)$ be the differential operator associated with the original equation (1.1) we are to solve, i.e., $Lu = 0$. The application of this operator is to be understood in the "weak" (or distributional) sense. Then,

$$\begin{aligned}
\mathcal{E} &= \int_{Q_T} L^*\,(L^*)^{-1}\,j(\mathbf{x},t)\ \ (u(\mathbf{x},t) - u_h(\mathbf{x},t))\,dx\,dt \\
&= \int_{Q_T} (L^*)^{-1}\,j(\mathbf{x},t)\ L\,(u(\mathbf{x},t) - u_h(\mathbf{x},t))\,dx\,dt \\
&= -\int_{Q_T} (L^*)^{-1}\,j(\mathbf{x},t)\ Lu_h(\mathbf{x},t)dx\,dt,
\end{aligned}$$

where L^* denotes the formal *adjoint* of L. This operator has the same form as L, but is to be understood as the operator in a wave equation running backwards in time. If u^* is the solution of the dual wave equation

$$L^*u^*(\mathbf{x},t) = j(\mathbf{x},t), \tag{3.8}$$

we get the following error identity

$$\mathcal{E} = -\int_{Q_T} u^*(\mathbf{x},t)\ Lu_h(\mathbf{x},t)\ dx\,dt. \tag{3.9}$$

Here, Lu_h is a measure for the numerical approximation of the equation (if u_h were the exact solution, then this expression would be zero), while u^* is a weight by which a certain space-time point enters the evaluation of the target functional.

If we knew the exact dual solution u^*, we had an exact representation of the error with respect to our target functional $J(\cdot)$, since the action of the differential operator L on the numerical solution is computable in form of cell-wise residuals. In some cases, the solution u^* of (3.8) can be computed exactly; for example if $J(u) = u(\mathbf{x}_0, T)$, i.e., $j(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_0)\delta(t-T)$, the exact dual solution is the advanced Green's function which is known analytically for constant coefficients and some domains (see Butkovski[7]). However, in most cases the dual solution will not be exactly computable, so we will need to replace it by a numerical approximation.

Finally, we mention that (3.9) allows for the required localization of the error contribution by the following representation,

$$\mathcal{E} = -\sum_{n=1}^{N}\sum_{K\in\mathbb{T}^n}\int_{I_n}\int_K u^*(\mathbf{x},t)\ Lu_h(\mathbf{x},t)dx\,dt, \tag{3.10}$$

containing the error contributions of all space-time cells $K \times I_n$.

3.2. *Formal derivation*

In this subsection, we give a mathematically rigorous derivation of the error identity (3.9), especially taking into account that we want to solve the equation in a velocity-displacement formulation. We will use the characteristic orthogonality property of Galerkin finite element methods to obtain a better localization of the error than in (3.10). Furthermore, we insert a step to get rid of the unspecified behavior of the second derivatives within L on the numerical solution u_h which in general has discontinuous gradients at interfaces between cells and discontinuous time derivatives at discrete time levels.

We first note that due to (1.2) and its discrete form (2.6), the error $\mathbf{e} = \mathbf{w} - \mathbf{w}_h$ satisfies an orthogonality relation with respect to the bilinear form $a(\cdot,\cdot)$:

$$a(\mathbf{e}, \boldsymbol{\tau}_h) = 0, \quad \boldsymbol{\tau}_h \in \mathcal{T}_h. \tag{3.11}$$

This property of finite element methods is called "Galerkin orthogonality".

For deriving our error estimate, we now define a continuous dual problem, corresponding to (3.8): Find $\mathbf{w}^* = (u^*, v^*) \in W$, such that

$$a^*(\mathbf{w}^*, \boldsymbol{\tau}) = J(\boldsymbol{\tau}) \quad \forall \boldsymbol{\tau} \in T,$$

with the dual bilinear form

$$a^*(\mathbf{w}^*, \boldsymbol{\tau}) = -(\rho \mathbf{w}_t^*, \boldsymbol{\tau})_{Q_T} + \left(\begin{pmatrix} 1 & 0 \\ 0 & \nabla \end{pmatrix} \mathbf{w}^*, \begin{pmatrix} 0 & -\rho \\ a\nabla & 0 \end{pmatrix} \boldsymbol{\tau} \right)_{Q_T} + (\rho \mathbf{w}^*(T), \boldsymbol{\tau}(T))_\Omega .$$

Note that if $\boldsymbol{\tau} \in W$ we get $a^*(\mathbf{w}^*, \boldsymbol{\tau}) = a(\boldsymbol{\tau}, \mathbf{w}^*)$ by partial integration with respect to the time variable. If we therefore test with $\boldsymbol{\tau} = \mathbf{e} \in W$ and use Galerkin orthogonality, we obtain

$$\mathcal{E} = J(\mathbf{e}) = a(\mathbf{e}, \mathbf{w}^* - \mathbf{w}_h^*) \tag{3.12}$$

for any $\mathbf{w}_h^* \in \mathcal{T}_h$. Recalling the definition of the bilinear form $a(\cdot, \cdot)$ and integrating by parts on each cell, we obtain the localized error identity

$$\mathcal{E} = \sum_{n=0}^{N} \sum_{K \in \mathbb{T}^n} \mathcal{E}_{K,n}, \tag{3.13}$$

with the cell-wise error indicators

$$\begin{aligned} \mathcal{E}_{K,n} &= -(r_1, u^* - u_h^*)_{K \times I_n} - (r_2, v^* - {v_h}^*)_{K \times I_n} - \tfrac{1}{2} (\mathbf{n} \cdot r_{\partial K}, v^* - v_h^*)_{\partial K \times I_n} , \\ \mathcal{E}_{K,0} &= (\rho(\mathbf{w}_0 - \mathbf{w}_h(0)), \mathbf{w}^*(0) - \mathbf{w}_h^*(0))_K , \end{aligned}$$

where $r_1 = \rho u_{h,t} - \rho v_h$ and $r_2 = \rho v_{h,t} - \nabla \cdot a \nabla u_h$ denote the cell residuals of the two equations and $\mathbf{n} \cdot r_{\partial K} = \mathbf{n} \cdot [a \nabla u_h]$ is the jump of the conormal derivative between two adjacent cells; these quantities are computable. Eq. (3.13) is analogous to (3.10) apart from the arbitrary element $\mathbf{w}_h^*$ which we could insert, and the integration by parts which we performed.

Eq. (3.13) makes reference to the unknown continuous dual solution $\mathbf{w}^*$. Hence, in evaluating it, we have to replace $\mathbf{w}^*$ by a suitable numerical approximation. For this, there are several possible methods[2]. We chose to compute $\mathbf{w}^*$ by higher order elements; this choice is motivated by academic reasons and is obviously not the way to go if computational resources are limited. There are other possibilities, however, by which the computation of the numerical dual solution takes only similar or less work than the primal problem; the latter is especially true for nonlinear problems where the dual problem is a linear one.

3.3. *Simplifications. Energy error indicators*

Sometimes, the target functional one is interested in is sufficiently global such that its domain of influence (which is given by the support of the dual solution) is more or less the whole domain and additionally the dual solution itself is smooth. Then one does not gain much by solving for the dual solution and one can get cheaper error indicators than the one above by using analytical *a priori* estimates for the dual solution.

The analysis for these cases is as follows: take the last term in (3.13) and estimate it using Cauchy-Schwarz's inequality, the interpolation estimate by Bramble and Hilbert and

a trace inequality:

$$\begin{aligned}
&\left(\mathbf{n}\cdot r_{\partial K}, v^* - v_h^*\right)_{\partial K\times I_n} \\
&\qquad \le \|\mathbf{n}\cdot[a\nabla u_h]\|_{\partial K\times I_n} \|v^* - {v_h}^*\|_{\partial K\times I_n} \\
&\qquad \le k_n^{1/2} \left\|\mathbf{n}\cdot[a\nabla(u_h^n + u_h^{n-1})]\right\|_{\partial K} C_I h^{-1/2} \left(h\|\nabla v^*\|_{K\times I_n} + k\|\partial_t v^*\|_{K\times I_n}\right).
\end{aligned}$$

For v_h^*, we have chosen the nodal interpolant of v^* and we have here assumed $r = 1$, i.e. linear elements. In some cases, we can further estimate using Schwarz's inequality and using the *a priori* estimates $\|\nabla v^*\|_{\Omega\times I_n} \le C_S$ and $\|\partial_t v^*\|_{\Omega\times I_n} \le C_S$ with a stability constant C_S; such estimates are often reasonable if the target functional is chosen such that the error to be bounded is $\mathcal{E} = J(e) = \left(\left\|\sqrt{\rho}(v - v_h)(\cdot,T)\right\|_\Omega^2 + \|\sqrt{a}\nabla(u - u_h)(\cdot,T)\|_\Omega^2\right)^{1/2}$, or some L_2 norms of the error. Though we know that in the first case $C_S \equiv 1$, sharp estimates for C_S are difficult to obtain in most other cases.

Similar estimates can be given for the first two terms in (3.13). However, it is known that for elements of odd order r, the interface residuals $\|\mathbf{n}\cdot r_{\partial K}\|_{\partial K}$ dominate the two cell residuals $\|r_{1,2}\|_K$, so for these elements

$$\eta_{K,n} = h^{1/2}\left\|\mathbf{n}\cdot[a\nabla(u_h^n + u_h^{n-1})]\right\|_{\partial K} \tag{3.14}$$

is a reasonable criterion for spatial refinement if k is small enough. Since the interpolation and particularly the stability constants C_I and C_S are unknown in general, we cannot estimate the error quantitatively, but the produced meshes are well suited for the approximation of the global solution. We will refer to (3.14) by the expression "energy error indicator". It should be noted that this error indicator is similar to the one derived by Kelly *et al.*[8] for Laplace's equation.

4. Algorithmic aspects

The mathematical ideas explained before are implemented in a program based on the `deal.II` finite element library[9] written in `C++` and which simulates wave propagation in one, two and three spatial dimensions. Its basic structure is along the diagram shown in Fig. 1. This basic algorithm will be detailed in the following.

Primal problem The solution of the forward problem is performed as a usual time stepping scheme. However, due to the fact that the computation of error estimates and maybe also the solution of the dual problem (if the error functional is nonlinear) relies on the knowledge of the primal solution, after solving each time step, grid refinement cannot be performed at once. Rather, we have to save the solution for future use; since for fine grids the memory needed by the solutions of all time steps can be in the order of several gigabytes, we have to store them onto disk rather than keep them in the main memory. The same holds for the grids which are rebuilt at the beginning of each time step (when solving primal and dual problem as well as when postprocessing) and stored on disk in the meantime.

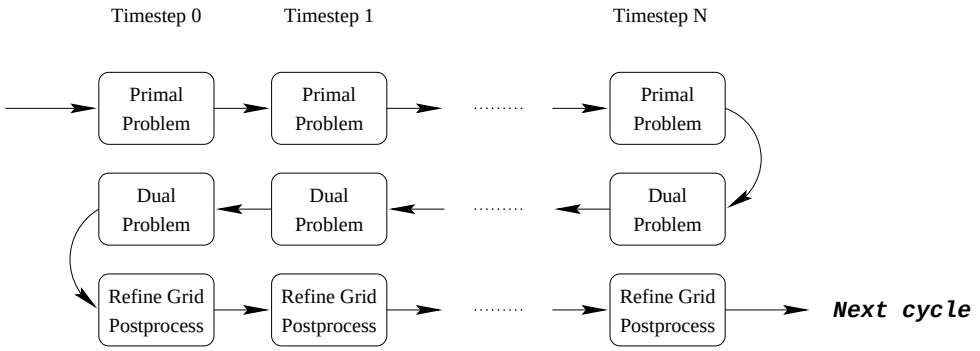

Fig. 1. One cycle of the basic algorithm for refinement by the dual estimator.

Dual problem The dual problem, being a wave equation to be solved backward in time, can be treated in the same way as the forward problem. In fact, most of the code is shared with the computation of the primal problem, even though the computations may use different time and space discretizations. In the present implementation, primal and dual problem are solved on the same space-time grid, which is not mandatory, however.

Postprocessing When the solutions of primal and dual problem are available, the error estimate (3.13) can be evaluated. Typically, this is also the time to evaluate the solution, e.g. to compute synthetic seismograms, evaluate functionals and get other data; we also generate graphical output at this stage.

After the error indicators for each space-time cell are stored and the other steps above have been performed, we can delete the temporary files holding primal and dual solution for the present time step. It has proven not worth the effort to keep and use them as starting values for the iterative solution of the systems of equations at the same time step in the next refinement cycle.

Grid refinement When all error indicators have been computed, the grids on the different time steps can be adapted. This is done by selecting either a number of space-time cells in the space-time triangulation based on their error indicator, or by refining each time step separately. Coarsening of cells is also possible. In future extensions of the program, we plan to implement an adaptive choice of time steps as well.

The exact details of grid refinement are complicated and in fact one of the most complex parts of the program. This is due to two reasons:

- *Refinement criterion:* two commonly used criteria for the selection of cells for refinement are a fixed fraction of cells with the largest refinement indicator, or those cells with highest indicator which make up a fixed fraction of the total error[10]. The actual performance of the algorithm depends strongly on the choice of one of these possibilities and on the mentioned fractions.

  When considering an outward traveling wave, the first strategy is obviously not suitable for individual refinement of time steps, since it does not allow a significantly

growing number of cells as the wave covers a larger region, if the fraction of cells is not increased for later time steps. This strategy is therefore only useful when refinement indicators are compared across time steps (which for present computations includes the task of comparing up to 100,000,000 indicators) in which case the numbers of cells to be refined or coarsened are usually chosen such that the total number of cells grows by a factor between two and three in each cycle.

In order to avoid the global comparison of error indicators, we usually use the second strategy, together with separate refinement of time steps. Since the region where the error is concentrated is usually rather small compared to the whole domain, a common value for the fraction of the total error that results in refinement is between 90 and 98 per cent, while the fraction resulting in coarsening is usually chosen to be half of the rest. In most cases, these values yield a doubling or tripling of the number of cells between subsequent cycles.

- *Diverging cell numbers:* Regardless of the refinement criterion chosen, the number of cells on subsequent time steps varies greatly. Without special treatment, the numbers can differ by a factor of two or more, therefore invalidating accurate computations on time steps with finer grids when going to a time step with a coarser grid.

  Algorithms have been developed for this problem[1] that adjust the refinement criterion on each time step iteratively and also perform some kind of mesh smoothing.

These problems, obviously, are shared by all implementations of residual-based adaptive algorithms for time dependent problems.

Splitting into cycles By a "cycle", we denote a complete set of solving forward and backward problem, postprocessing and grid refinement. These steps on subsequently finer grids are performed in an iteration until the desired accuracy or the limit of computational resources is reached. Due to the increase in numbers of cells, every cycle typically uses at least twice as much computing time and memory than the previous one, so the total cost is always dominated by the computations on the last grids and the iterative refinement is a justifiable effort. However, for this we have to refine rather crudely, i.e., the cautious refinement leading to optimal meshes applicable to stationary problems like Laplace's equation[11] is not possible here.

Simplifications when using energy error indicators When using energy-error indicators instead of the dual estimator, no global dual problem needs to be solved and the operations listed under "postprocessing" can be done after computing each time step of the primal problem. We then need only one loop over all time steps within each cycle if error indicators are not to be compared across different time steps.

A further simplification can be achieved if the stopping criterion for the number of cycles is not convergence to a specified level of error but simply a fixed number of cycles. We can then rewrite the algorithm as shown in Fig. 2. In this case, no intermediate solutions

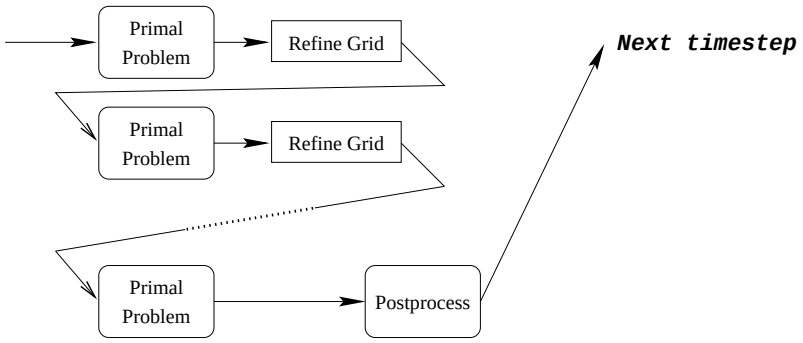

Fig. 2. Simplified algorithm for refinement by the energy error indicator.

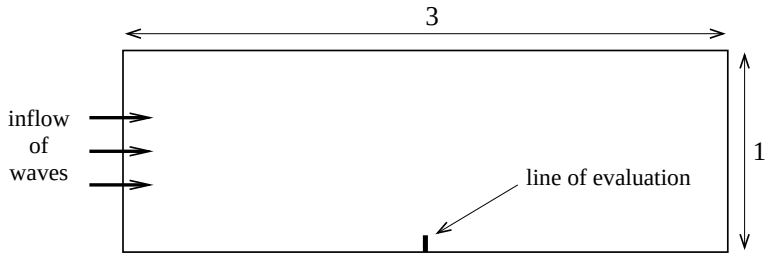

Fig. 3. Configuration of the domain in example 1.

need to be stored and the algorithm closely resembles those used for adaptive ODE solvers, for example; this scheme is at present also the most commonly used one in solving time dependent partial differential equations (see for example reference [12], and the references therein).

5. Numerical results

Example 1 Let $\Omega = (0,3) \times (0,1)$, and let the initial and boundary conditions be chosen such that we have waves propagating from the left into the domain (see Fig. 3):

$$\begin{aligned} u_0 = v_0 &= 0, \\ u(\mathbf{x},t) &= \left(\sin \frac{5\pi t}{2}\right)^2 && \text{for } \mathbf{x} \in \partial\Omega \cap \{x = 0\}, \\ \partial_n u &= 0 && \text{for } \mathbf{x} \in \partial\Omega \backslash \{x = 0\}. \end{aligned}$$

When solving eq. (1.1) with $a = \rho = 1$ with these data, and using adaptive grids refined on the basis of the energy error indicator (3.14), we obtain meshes as shown in Fig. 4. The meshes closely match the propagating wave and are coarse in front of it where the solution is smooth.

However, we might be interested in a part of the wave field in a certain time window only. For this example, we chose a strip at $x = 1.5, 0 \le y \le \frac{1}{16}$, and the time interval is

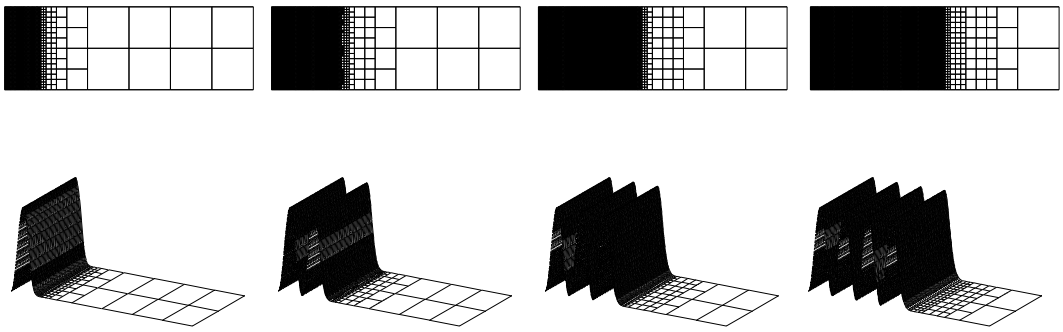

Fig. 4. Results for example 1, with refinement based on the energy error indicator (3.14). Top row: grids after three refinement cycles at times $t = 0.4, 0.8, 1.2$ and 1.6; bottom row: primal solution u at the same times.

chosen such that it is centered around the arrival time of the peak of the first wave which is traveling from the left. A suitable functional describing this error measure would be

$$J(\mathbf{w}) = \int_{1.6}^{1.8} \int_{0}^{\frac{1}{16}} u_{|x=1.5}(\cdot, y, t)\, dy\, dt.$$

If we solve the respective dual problem and refine according to (3.13), we obtain meshes as shown in Fig. 5. In the bottom row, the dual solution is shown, which here roughly is a half-circular wave traveling backwards in time (it is the convolution of the advanced Green's function with the integral kernel of J). Refinement occurs where the dual solution is large and where the residuals are large, i.e. mainly in the region where waves presently are. Therefore, only a very small part of the wave field and the dual solution as well is resolved by the grid. The rest of the wave field, especially the regions behind the first wave and those that pass by the region of evaluation are neglected and, indeed, are hardly identifiable compared with the well-resolved wave field in Fig. 4.

Example 2 We consider the propagation of an outward traveling wave on $\Omega = (-1, 1)^2$ with a strongly distorted coefficient. Layout of the domain and structure of the coefficient are shown in Fig. 6. Boundary and initial conditions were chosen as follows:

$$\mathbf{n} \cdot a\nabla u = 0 \quad \text{on } y = 1, \qquad u = 0 \quad \text{on } \partial\Omega \backslash \{y = 1\},$$

$$u_0 = 0, \qquad v_0 = \theta(s - r)\exp\left(-\frac{|\mathbf{x}|^2}{s^2}\right)\left(1 - \frac{|\mathbf{x}|^2}{s^2}\right),$$

with $s = 0.02$ and $\theta(\cdot)$ the jump function. The region of origin of the wave field is significantly smaller than shown in Fig. 6. Notice that the lowest frequency in this initial wave field has wave length $\lambda = 4s$; hence taking the common twenty grid points per wave length would yield $250,000$ cells already for the largest wave length. Uniform grids obviously quickly get to their limits in such cases.

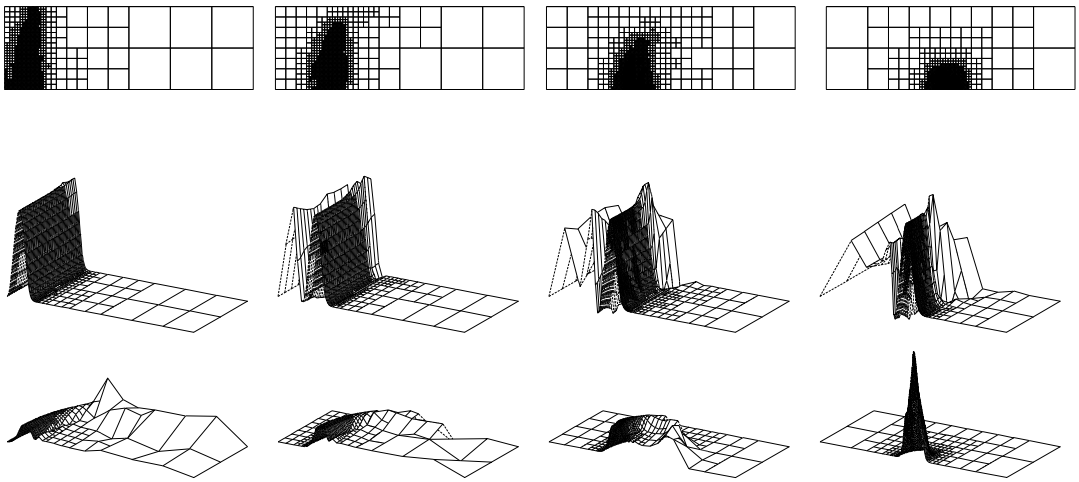

Fig. 5. Results for example 1, with refinement by the dual estimator (3.13). Top row: computational grids after four refinement cycles; middle: primal solution u, bottom row: dual solution $\bar{v}$.

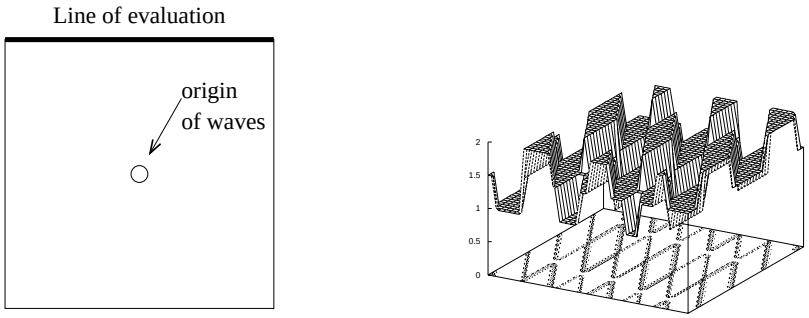

Fig. 6. Layout of the domain (left) and structure of the coefficient $a(\mathbf{x})$ (right) for example 2.

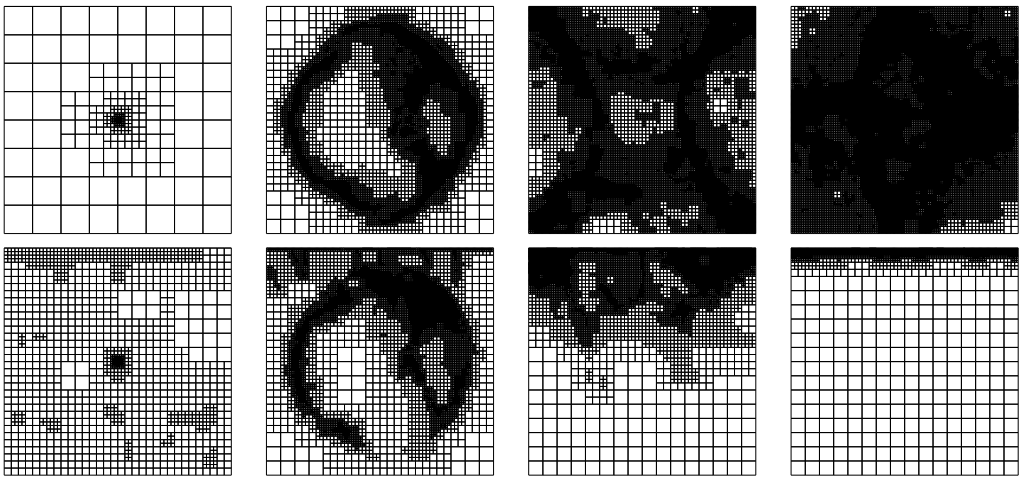

Fig. 7. Results of example 2. Grids produced from refinement by the energy error indicator (3.14) (top row) and by the dual estimator (3.13) (bottom row) at times $t = 0, \frac{2}{3}, \frac{4}{3}$ and $t = T = 2$.

If we consider this example as a model of propagation of seismic waves in a faulted region of rock, then we would be interested in recording seismograms at the surface, which we here chose as the top line of the domain. Since our approach only allows us to select a scalar value as quantity of interest, we have to choose a functional which is sensitive to changes in the solution at all times and along the whole line of evaluation; one such functional is

$$J(\mathbf{w}) = \int_0^T \int u_{|y=1}(x, \cdot, t)\ \omega(x,t)\ dx\, dt$$

with a weight factor $\omega(x,t) = \sin(3\pi x) \sin(5\pi t/T)$. The frequency of oscillation of this weight is chosen to match the frequencies in the wave field to obtain good resolution of changes.

In Fig. 7 we show the grids resulting from refinement by the energy error indicator (3.14) and the dual estimator (3.13). The first one resolves the wave field well, including reflections from discontinuities in the coefficient. The second additionally takes into account, that the lower parts of the domain lie outside the domain of influence of the target functional if we truncate the time domain at $T = 2$; this domain of influence constricts to the top as we approach the final time, as is reflected by the produced grids.

A seismogram, taken at the points at the top of the domain after each $\delta t = 0.02$, is shown in Fig. 8. The distorted structure is due to the rough coefficient and its oblique lines of discontinuity. In the preparation of this seismogram, we have taken the data from the simulation on the grids refined by the dual estimator (3.13) (bottom row of Fig. 7). There is no discernible difference to the seismograms obtained with the help of grids produced by (3.14) (top row of Fig. 7). In fact, the difference is on the same level as the changes observed when using a once-more or once-less refined grid; the difference in results between the two methods of refining the grids is thus in the same range as the discretization error, although the grids generated by (3.13) use significantly less cells.

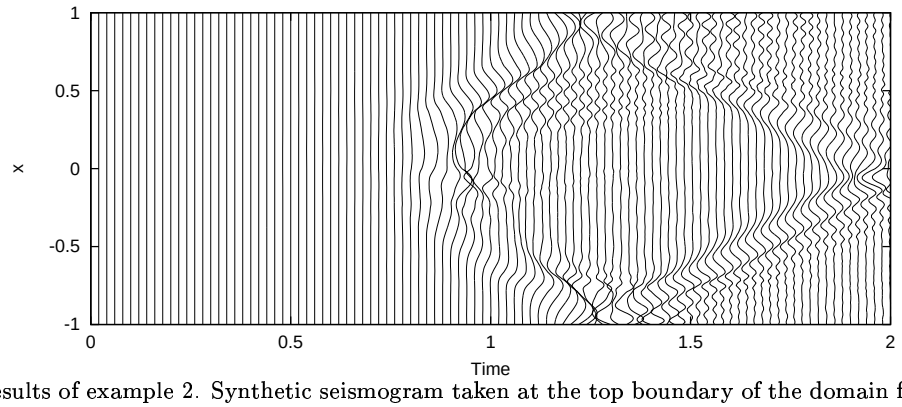

Fig. 8. Results of example 2. Synthetic seismogram taken at the top boundary of the domain for refinement with the dual estimator (3.13).

6. Open problems

At present, the method described in this work still suffers from two defects which limit its ability to outperform energy-error-indicator-based mesh refinement by a larger factor than presented. The first one is a defect of the discretization while the origin of the second one is not yet fully understood. We briefly comment on these defects.

Spurious reflection from grid discontinuities It is well known that waves traveling from a region with a fine grid to one with a coarse grid suffer from spurious reflection of high frequency parts at the grid interface. This is not a problem when using the energy error indicator (3.14), because then the grid will be refined at the locations of the waves and no major wave will ever hit a strong discontinuity of the mesh size. It is a problem, however, if the dual estimator (3.13) is used because it includes information about the domain of dependence of the target functional and outside of this domain the grid will be made very coarse. However, there will still be waves traveling into these regions which then will be partly reflected. The error estimator "sees" this and refines the grid accordingly to reduce the influence of these reflections on the target functional, but refinement can be slow in these regions. It would be desirable to reduce these spurious reflections by modifying the discretization in order to make the approach work even better.

The problem of spurious reflections is well known for a long time.[13,14] Some strategies have been suggested to reduce them for the Helmholtz equation,[15] but no results seem to be available for the wave equation in time domain.

Unreliable error estimates In most cases, (3.12) and (3.13) do not yield reliable error bounds unless the time step size is chosen rather small. They may be one or two orders of magnitude away from the true size of the error and may not show a monotone convergence behavior. It should be noted, however, that the localization of the error is very good, so the resulting grids closely match what one would expect and are very efficient for the task given; the lack of reliability of the quantitative behavior of the error estimator therefore

does not reduce its capability as a refinement indicator. Also,

$$\eta = \sum_{n=0}^{N} \sum_{K \in \mathbb{T}^n} |\mathcal{E}_{K,n}|$$

with the cell-wise quantities $\mathcal{E}_{K,n}$ taken from (3.13) reliably bounds the true error.

The reasons for this behavior are not totally clear at present but are believed to originate in the replacement of the exact dual solution $\mathbf{w}^*$ in (3.13) by a numerically obtained approximation. Further investigations are presently underway to determine the cause and a remedy to this problem.

7. Conclusion

A new approach to the adaptive solution of the acoustic wave equation has been presented which furnishes control of the error in quantities of physical interest. It is based on weighted *a posteriori* error estimates which are obtained by solving a global dual problem.

An algorithm has then been described which uses the *a posteriori* error estimates to generate a sequence of successively refined computational meshes until a desired accuracy is reached or the computational resources are exhausted. This algorithm differs significantly from the usual manner in which adaptive algorithms are used, as it iterates the solution over the whole space-time domain rather than separately on each single time step.

Two examples show how the method works and demonstrate the superiority of the produced grids over the ones generated by energy error indicators. The reason is that taking into account a dual problem allows to restrict mesh refinement to those regions lying inside the domain of influence of the quantity of interest; hence the area of refinement may be significantly smaller than the whole time-space domain.

Finally, two of the major open problems with the presented approach have been addressed. They stem from inaccurate discretization and problems with the evaluation of the error identity (3.13). Solving these problems is subject of ongoing work.

Acknowledgments

The authors acknowledge the support by the German Research Association (DFG) through the Graduiertenkolleg and the SFB 359 at the IWR, Universität Heidelberg.

References

1. W. Bangerth, "Adaptive Finite-Elemente-Methoden zur Lösung der Wellengleichung mit Anwendung in der Physik der Sonne," Diplomarbeit, Institut für Angewandte Mathematik, Universität Heidelberg, 1998.
2. R. Becker and R. Rannacher, *East–West J. Num. Math.*, **4**, 237–264 (1996).
3. W. Bangerth and R. Rannacher, *East–West J. Numer. Math.*, **7**, 263–282 (1999).
4. D. A. French and T. E. Peterson, *Math. Comp.*, **65**, 491–506 (1996).
5. R. Becker and R. Rannacher, "Weighted a posteriori error control in FE methods," *Proceedings of ENUMATH 95, in Proceedings of ENUMATH 97* (H. G. Bock *et al.*, eds.), World Scientific, 1998.

6. K. Eriksson and C. Johnson, *Math. Comp.*, **50**, 361–383 (1988).
7. A. G. Butkovski, *Green's Functions and Transfer Functions Handbook*. Ellis Horwood Ltd., 1982.
8. D. W. Kelly, J. R. Gago, O. C. Zienkiewicz, and I. Babuška, *Internat. J. Numer. Methods Engrg.*, **19**, 1593–1619 (1983).
9. W. Bangerth and G. Kanschat, "Concepts for object-oriented finite element software – the `deal.II` library," preprint 99-43, IWR Heidelberg, Oct. 1999. See also `http://gaia.iwr.uni-heidelberg.de/~deal/`.
10. G. Kanschat, "Parallel and Adaptive Galerkin Methods for Radiative Transfer Problems," Dissertation, Universität Heidelberg, 1996.
11. R. Becker and R. Rannacher, "A general concept of adaptivity in finite element methods with applications to problems in fluid and structural mechanics," in *Grid Generation and Adaptive Algorithms* (M. Bern, J. E. Flaherty, and M. Luskin, eds.), vol. 113 of *IMA Volumes in Mathematics and its Applications*, Springer, 1998.
12. X. D. Li and N.-E. Wiberg, *Comput. Methods Appl. Mech. Engrg.*, **156**, 211–229 (1998).
13. Z. P. Bažant, *Comput. Methods Appl. Mech. Engrg.*, **16**, 91–100 (1978).
14. I. Freid, *Comput. Methods Appl. Mech. Engrg.*, **20**, 317–321 (1979).
15. I. Harari, *Comput. Methods Appl. Mech. Engrg.*, **140**, 39–58 (1997).